## AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions and listings of claims in the application.

**LISTING OF CLAIMS**

1. (Original) An embedded disk controller having a servo controller, comprising:

a servo controller interface with a speed matching module and a pipeline control module such that at least two processors share memory mapped registers without conflicts.

2. (Original) The controller of Claim 1, where one processor operates at a first frequency and a second processor operates at the second frequency.

3. (Original) The controller of Claim 1, where the servo-controller and the servo controller interface operate in same or different frequency domains.

4. (Original) The controller of Claim 1, the speed matching module ensures communication without inserting wait states in a servo controller interface clock domain for write access to the servo controller.

5. (Original) The controller of Claim 1, where there is no read conflicts between the first and second processor.

6. (Original) The controller of Claim 1, provides a hardware mechanism for indivisible register access to the first or second processor.

7. (Original) The controller of Claim 6, where the hardware mechanism includes a hard semaphore.

~~9~~ 8. (Currently Amended) The controller of Claim 6, where the hardware mechanism includes a soft semaphore.

~~10~~ 9. (Currently Amended) The controller of Claim 1, where the pipeline control module resolves conflict between the first and second processor transaction.

~~11~~ 10. (Currently Amended) The controller of Claim 1, where the first and second processor communicate with the servo controller via two separate buses.

~~12~~ 11. (Currently Amended) The controller of Claim 1, where if there is a write conflict between the first and second processor, pipeline control module holds write access to the second processor.

~~13~~ 12. (Currently Amended) The controller of Claim 6, where the hardware mechanism is a semaphore register.

~~14~~ 13. (Currently Amended) A system for reading and writing data to a storage medium, comprising:

an embedded disk controller having a servo controller interface module that includes a speed matching module and a pipeline control module such that at least two processors share memory mapped registers without conflicts.

~~15~~ 14. (Currently Amended) The system of Claim ~~14~~ 13, where one processor operates at a first frequency and a second processor operates at the second frequency.

~~16~~ 15. (Currently Amended) The system of Claim ~~14~~ 13, where the servo-controller and the servo controller interface operate in same or different frequency domains.

~~17~~ 16. (Currently Amended) The system of Claim ~~14~~ 13, the speed matching module ensures communication without inserting wait states in a servo controller interface clock domain for write access to the servo controller.

~~18~~ 17. (Currently Amended) The system of Claim ~~14~~ 13, where there is no read conflicts between the first and second processor.

~~19~~ 18. (Currently Amended) The system of Claim ~~14~~ 13, provides a hardware mechanism for indivisible register access to the first or second processor.

~~20~~ 19. (Currently Amended) The system of Claim ~~19~~ 18, where the hardware mechanism includes a hard semaphore.

~~21~~ 20. (Currently Amended) The system of Claim ~~19~~ 18, where the hardware mechanism includes a soft semaphore.

~~22~~ 21. (Currently Amended) The system of Claim ~~14~~ 13, where the pipeline control module resolves conflict between the first and second processor transaction.

~~23~~ 22. (Currently Amended) The system of Claim ~~14~~ 13, where the first and second processor communicate with the servo controller via two separate buses.

~~24~~ 23. (Currently Amended) The controller system of Claim ~~14~~ 13, where if there is a write conflict between the first and second processor, pipeline control module holds write access to the second processor.

~~25~~ 24. (Currently Amended) The system of Claim ~~19~~ 18, where the hardware mechanism is a semaphore register.

25. (New) A servo controller interface for a disk controller, comprising:

a first interface for communicating with a first processor over a first bus; and

a second interface for communicating with a second processor over a second bus,

wherein the servo controller interface selectively grants one of the first and second processors access to a servo controller.

26. (New) The servo controller interface of claim 25 wherein the first processor operates at a first frequency and the second processor operates at a second frequency.

27. (New) The servo controller interface of claim 25 further comprising a speed matching module that resolves conflicts between at least first and second clock domains.

28. (New) The servo controller interface of claim 27 wherein the speed matching module transitions servo controller accesses from one of the first and second clock domains to the other of the first and second clock domains.

29. (New) The servo controller interface of claim 25 wherein the first and second processors share memory mapped registers within the servo controller.

30. (New) The servo controller interface of claim 27 wherein the speed matching module does not insert wait states in a clock domain of the servo controller interface during write access to the servo controller.

31. (New) The servo controller interface of claim 25 further comprising a pipeline control module that resolves transaction conflicts between the first processor and the second processor.

32. (New) The servo controller interface of claim 25 wherein the servo controller interface delays a write access for one of the first and second processors during write conflicts between the first and second processors.

33. (New) A servo controller interface for a disk controller, comprising:

first interface means for communicating with a first processor over a first bus; and

second interface means for communicating with a second processor over a second bus,

wherein the servo controller interface selectively grants one of the first and second processors access to a servo controller.

34. (New) The servo controller interface of claim 33 wherein the first processor operates at a first frequency and the second processor operates at a second frequency.

35. (New) The servo controller interface of claim 33 further comprising speed matching means for resolving conflicts between at least first and second clock domains.

36. (New) The servo controller interface of claim 35 wherein the speed matching means transitions servo controller accesses from one of the first and second clock domains to the other of the first and second clock domains.

37. (New) The servo controller interface of claim 33 wherein the first and second processors share memory mapped registers within the servo controller.

38. (New) The servo controller interface of claim 35 wherein the speed matching module does not insert wait states in a clock domain of the servo controller interface during write access to the servo controller.

39. (New) The servo controller interface of claim 33 further comprising pipeline control means for resolving transaction conflicts between the first processor and the second processor.

40. (New) The servo controller interface of claim 33 wherein the servo controller interface delays a write access for one of the first and second processors during write conflicts between the first and second processors.

41. (New) A method for operating a servo controller interface having a first interface and a second interface, comprising:

communicating with a first processor over a first bus using the first interface;

communicating with a second processor over a second bus using the second interface; and

selectively granting one of the first and second processors access to a servo controller with the servo controller interface.

42. (New) The servo method of claim 41 wherein the first processor operates at a first frequency and the second processor operates at a second frequency.

43. (New) The method of claim 41 further comprising resolving conflicts between at least first and second clock domains at a speed matching module.

44. (New) The method of claim 43 further comprising transitioning servo controller accesses from one of the first and second clock domains to the other of the first and second clock domains at the speed matching module.

45. (New) The method of claim 41 wherein the first and second processors share memory mapped registers within the servo controller.

46. (New) The method of claim 41 further comprising resolving transaction conflicts between the first processor and the second processor at a pipeline control module.

47. (New) The method of claim 41 further comprising delaying a write access for one of the first and second processors during write conflicts between the first and second processors.